

APLICACIÓN MÓVIL PARA FUMIGACIÓN SEMI-AUTÓNOMA EN INVERNADEROS

*M.C. LARISSA JEANETTE PENICHE RUIZ*¹

*Br. JESÚS BENJAMÍN ARAGÓN SOLÍS*²

*Br. ABRAHAM GUERRERO GODOY.*³

RESUMEN

Diseñar, desarrollar e implantar un sistema de fumigación semi-automática por medio de un vehículo terrestre no tripulado, controlado por un dispositivo móvil para repeler plagas que afectan a los invernaderos. Esto podrá brindar mayor control de los invernaderos y de los tiempos de fumigación. Se puede adaptar de acuerdo a las necesidades de los invernaderos y modificar el tipo de fumigación dependiendo del tipo de plaga que se requiera controlar. Se configura un mapeo de la zona a fumigar, permitiendo cierto grado de independencia del vehículo y reducida interacción física del usuario con las sustancias de fumigación.

CONCEPTOS CLAVE: APLICACIÓN MÓVIL, FUMIGACION SEMI-AUTÓNOMA, PLAGAS DE INVERNADEROS

¹ Maestro en Ciencias Computacionales, Instituto Tecnológico de Mérida, larissa.peniche@itmerida.mx

² Bachiller, Instituto Tecnológico de Mérida, benjaminaragon94@gmail.com

³ Bachiller, Instituto Tecnológico de Mérida, abrahamgg@outlook.com

1. INTRODUCCIÓN.

Realizar un exhaustivo control de plagas es una cuestión irrenunciable para los agricultores. La proliferación de plagas puede provocar graves problemas, tanto desde el punto de vista económico como desde el bienestar físico, ya que pueden ser una fuente de problemas para la salud.

En los invernaderos proliferan distintas plagas que pueden llegar a ser un problema cuando se encuentran en gran cantidad y conllevan consecuencias presupuestarias. Las medidas para combatirlas generalmente implican gastos que incluyen los costos de inspección, seguimiento, prevención y reacción. Los costos de algunas de estas medidas son proporcionales al tamaño del sector afectado, ya sea el agrícola o incluso en el sector comercial.

Un ejemplo de esto es que hoy en día una fumigación presencial cuesta alrededor de 500 a 800 pesos para áreas de cultivo con medidas de 10m x 30m que son 300m²; los efectos tienden a tener de apenas unos pocos días de efectividad de la fecha de aplicación y cuentan con la desventaja de que los plaguicidas son dañinos para la salud humana (en mayor o menor medida) e incluso podrían dañar los mismos cultivos que se pretende proteger o recuperar de plagas. (Ver Figura 1) (SAGARPA, (n.dd), (OMS, 2015).

Otra forma es la instalación de sistemas para la fumigación, los cuales cumplen con el propósito y economizan costos a comparación de la fumigación presencial, pero tienen la desventaja de no ser transportable o de muy difícil reinstalación.

Figura 1. Fumigación presencial



1.2 Planteamiento del problema.

Los negocios del sector agrario que tienen problemas con plagas, recurren a la fumigación como proceso correctivo aunque la fumigación debiera ser usada como proceso preventivo. Sin embargo, en ocasiones los dueños dudan en contratar a las empresas fumigadoras, ya sea porque se sienten inseguros de que entren a sus plantaciones o invernaderos, personas desconocidas para realizar el trabajo; en otros casos no logran concordar un horario conveniente a ambas partes, ya que las distancias a los invernaderos y las diversas locaciones de los mismos pueden crear complicaciones al momento de las visitas de fumigación; y en el peor de los casos, el costo de contratar empresas especializadas en fumigación supera la capacidad económica del invernadero por lo que se ve en la necesidad de hacerlo por su cuenta.

1.3 Propuesta solución.

Se propone crear un sistema de fumigación autónoma por medio de un **carrito tipo vehículo terrestre no tripulado** (UGV), dirigido por medio de una aplicación móvil para dispositivo celular o Tablet que controle el proceso de fumigación de plagas que afectan a los invernaderos, con reducida interacción humana con los plaguicidas dañinos a la salud.

Se brindará un mayor control de sus invernaderos y los tiempos de fumigación; éste se adaptará de acuerdo a sus necesidades y al tipo de plaga que se quiera controlar. Se contará con una configuración de mapeo (trazo de ruta) de la zona a fumigar, permitiendo que sea casi independiente y la interacción del usuario sea mínima.

1.4 Objetivos.

1.4.1 General

Diseñar, desarrollar e implementar una aplicación móvil programable de fumigación por medio de un carrito o vehículo terrestre no tripulado UGV para el control de fumigaciones contra plagas en invernaderos.

1.4.2 Específicos

- Diseñar y programar una aplicación para dispositivo móvil que realice el manejo del sistema de control.
- Diseñar y desarrollar un vehículo capaz de solucionar inconvenientes de movilidad y transporte para el sistema de fumigación.
- Desarrollar el sistema con base a las normas, leyes y estándares de salubridad del país.
- Disminuir el contacto humano con los agroquímicos.
- Implementar los diversos tipos de repelentes dependiendo de la plaga a solucionar.

1.5 Justificación.

Los procesos agrícolas demandan eficiencia, tanto en su producción como en su calidad, el sistema autónomo móvil de fumigación puede coadyuvar a mejorar la calidad de los productos cosechados y reducir el tiempo y esfuerzo de fumigación. El control de plagas puede aumentar las cantidades del producto cosechado y además, utilizando un biocida de origen natural, los productos cosechados no serán dañinos para el consumo humano.

1.6 Delimitación.

1.6.1 Alcances

- Tener la comodidad de programar los horarios de fumigación vía celular.
- Controlar y repeler diversas plagas.
- Disminuir las enfermedades y alergias provocadas por los plaguicidas usados comúnmente en fumigación tradicional.
- Sentar las bases para impulsar una empresa que ocupe el nicho de dar servicios de fumigación en invernaderos de la localidad.

1.6.2 Limitaciones

- El sistema solo funcionará mientras sea alimentado por una fuente de energía eléctrica (batería o pila) y cuente con producto repelente.
- Será una medida preventiva y/o correctiva para las plagas según el caso lo amerite.
- El UGV solamente podrá llevar a cabo su función si cuenta con camino suficiente para poder movilizarse en el área.
- Desarrollado para sistemas Android 4.4 o versiones superiores.

2 ESTADO DEL ARTE

2.1 Rip Mosquito & Insects

Rip Mosquito & Insects es una empresa fundada en 2006 dedicada a la implementación y venta de sistemas automáticos para la eliminación de insectos y mosquitos en general (Rip Mosquito & insects, 2006).

Esta empresa desarrolló un sistema amigable al usuario y completamente automatizado (solo se necesita programar las horas dependiendo del cliente), mediante el uso de hardware propio

de la empresa, con uso de insecticida biodegradable también propio, que aniquila y repele insectos de forma inmediata, silenciosa y natural.

Teniendo su base en el Distrito Federal, han realizado trabajos en Acapulco en sitios como establos, edificios de profesionistas y casas domésticas.

El equipo que ellos implantan funciona inyectando líquido a gran presión que envía a unos aspersores de acero niquelado, los cuales se instalan en puntos estratégicos. Después de hecha la instalación, se activa el sistema y éste crea una finísima nebulización de acción prolongada que elimina y repele mosquitos, moscas, zancudos y otros insectos voladores y rastreros. Se puede activar mediante un control remoto.

2.2. Trakür

Es un robot presentado en Argentina, por el Instituto de Ingeniería Rural del Instituto Nacional de Tecnología Agropecuaria (por sus siglas, INTA) en julio del 2014 (Trakür , 2014). Diseñado para proteger la salud de los operarios reduciendo su exposición a los agroquímicos, también para el control de plagas en cultivos bajo cubierta.

Es un robot que cuenta con una alimentación de 12V, tiene una autonomía de 8 horas y permite transportar entre 90 a 100kg; es guiado por cable, cuenta con un motor eléctrico en la parte inferior el cual controla el sistema de movimiento y carga con el peso del robot; en la parte superior cuenta con una bomba centrífuga y un depósito que permiten esparcir líquidos que no sean igual o más viscosos que el aceite que posea este depósito. En la parte frontal y trasera el sistema de guiado se realiza mediante un cable enterrado o sobre la superficie que emite corriente eléctrica variable que utiliza como señal y son detectadas por los sensores del robot.

Calcula la distancia promedio entre los dos sensores que posee el robot, pudiendo avanzar con una trayectoria definida.

2.3 ByeBugs

Es una empresa nacida en Culiacán, Sinaloa, en el año 2007; habiendo realizado trabajos de fumigación previos, desarrollaron un sistema para el control de plagas de insectos voladores y rastreros y ofrecen sus servicios profesionales en este ramo tanto para el uso de sus sistemas como para fumigación tradicional (Bye Bugs, 2007).

El sistema automatizado que ofrecen se llama Fly Defense; el sistema efectúa la nebulización en tiempo y concentraciones programadas especialmente para el área y el tipo de plaga a combatir para que el espacio definido esté libre de insectos permanentemente o solo en los horarios que se necesite.

El mecanismo computarizado inyecta un líquido a gran presión a través de boquillas de alto rendimiento, previamente instaladas en puntos donde se requiere mayor protección. Esto crea una finísima nebulización de acción prolongada que elimina y repele mosquitos, moscas y otros insectos voladores en patios, terrazas, y áreas más amplias, como jardines, prados y canchas deportivas.

El sistema funciona de forma automática y también es posible activarlo mediante un control remoto. Utiliza una manguera para llegar a más de 500 metros con 115 boquillas las cuales pueden ser instaladas en cualquier lugar que sea necesario.

2.3.- Metodología de desarrollo RUP.

La metodología de desarrollo a utilizar para la elaboración del proyecto es la metodología RUP (Proceso Racional Unificado, por sus), que es una metodología de desarrollo de software formal, orientadas a objetos, con un ciclo de vida espiral.

Su meta principal es asegurar la producción de software de alta calidad que cumpla con las necesidades de los usuarios, con una planeación y presupuesto predecible.

La metodología RUP, divide en 4 fases el desarrollo del software. Cada Fase tiene definido un conjunto de objetivos y un punto de control específico, como se muestra en la Tabla 1.

Tabla 1. Objetivos de las fases de RUP y sus puntos de control.

Fase	Objetivos	Puntos de Control
Inicio	<ul style="list-style-type: none"> Definir el alcance del proyecto Entender que se va a construir 	Objetivo del proyecto
Elaboración	<ul style="list-style-type: none"> Construir una versión ejecutable de la arquitectura de la aplicación Entender cómo se va a construir 	Arquitectura de la Aplicación
Construcción	<ul style="list-style-type: none"> Completar el esqueleto de la Aplicación con la funcionalidad Construir una versión Beta 	Versión Operativa Inicial de la Aplicación
Transición	<ul style="list-style-type: none"> Poner a disposición la aplicación para los usuarios finales Construir la versión Final 	Liberación de la versión de la Aplicación

2.3 Arduino

Arduino es una plataforma de código abierto que se utiliza para la construcción de proyectos de electrónica. Esta plataforma, consiste tanto en un tablero de circuito físico programable (a menudo referido como un microcontrolador) y una pieza de software, o IDE (entorno de

desarrollo integrado) que se ejecuta en el ordenador, se utiliza para escribir y cargar el código de computadora a la tarjeta física (Arduino, 2007).

A diferencia de la mayoría de las tarjetas de circuitos programables, el Arduino no necesita un hardware independiente, con el fin de cargar el nuevo código en la tarjeta, más que utilizar un cable USB. Además, el IDE Arduino utiliza una versión simplificada de C++, por lo que es más fácil de aprender a programar. Por último, Arduino proporciona un factor de forma estándar que desata las funciones del microcontrolador en un formato adaptado.

Para fines de este proyecto, se usan tanto el hardware y el software de esta compañía para la comunicación entre los elementos.

Hardware Arduino

El Arduino “Uno” tiene un total de 14 pins de E/S, 6 de los cuales soportan entradas analógicas. Seis de los pines se puede utilizar como la modulación de ancho de pulso o salidas PWM (que simplemente se pone resultado análogo a través de medios digitales), una conexión USB y un conector de corriente. El Uno puede ser alimentado a través de USB, una fuente de energía externa o una batería.

Software Arduino

Para el desarrollo de aplicaciones para Arduino, la empresa distribuye su propio IDE para casi todas las plataformas (Windows, Linux, Mac). La ventaja principal es que al ser de carácter libre, no se paga ni una licencia ni costo adicional por el software.

Para la programación en este dispositivo y plataforma, se utiliza un lenguaje basado en C++ pero a su vez utiliza un concepto diseñado para los microcontroladores llamado Wiring, se dice

que su lenguaje es Processing/Wiring que se divide la aplicación de eventos producidos por los usuarios utilizando librerías mezclado con un entorno de entradas/salidas para el uso de dispositivos electrónicos (Reas C. and Fry B. , 2014).

2.4 Android

Android es un sistema operativo de código abierto basado en Linux para dispositivos móviles como teléfonos inteligentes y Tablets. Android fue desarrollado por la Open Handset Alliance, encabezada por Google y otras compañías. Android ofrece un enfoque unificado para el desarrollo de aplicaciones para dispositivos móviles. Esto ayuda a los desarrolladores, ya que sólo tienen que desarrollar para Android, y sus aplicaciones tendrán la capacidad de ejecutarse en diferentes dispositivos que usen Android (Android Developers, 2012).

Aplicaciones de Android

Las aplicaciones Android se desarrollan normalmente en lenguaje Java utilizando el Kit de desarrollo de software Android. Una vez desarrollado, las aplicaciones de Android pueden ser compiladas y vendidas fácilmente, ya sea a través de una tienda como Google Play, SlideME, Opera Mobile Store, Mobango, F-androide y la Appstore de Amazon.

2.5 Android Studio

Android Studio es un IDE (entorno de desarrollo integrado, por sus siglas en inglés), basado en IntelliJ IDEA de la compañía JetBrains. Este IDE proporciona varias mejoras con respecto al plugin ADT (Android Developer Tools) para Eclipse. Utiliza una licencia de software libre Apache 2.0, está programado en Java y es multiplataforma.

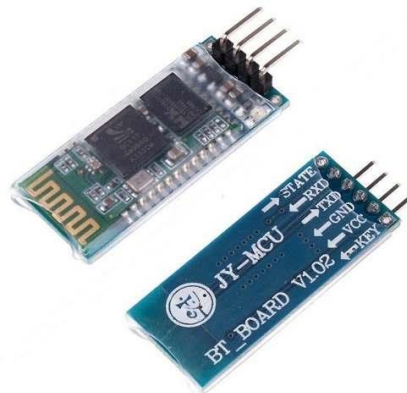
Fue presentado por Google el 16 de mayo del 2013 en el congreso de desarrolladores Google I/O, con el objetivo de crear un entorno dedicado en exclusiva a la programación de

aplicaciones para dispositivos Android, proporcionando a Google un mayor control sobre el proceso de producción.

2.6 Módulo Bluetooth

El Arduino BT es una placa electrónica que originalmente se basó en el ATmega168, pero ahora se suministra con el 328 y el módulo Bluegiga WT11 Bluetooth. Puede soportar una comunicación serial inalámbrica a través de Bluetooth (pero no es compatible con auriculares Bluetooth u otros dispositivos de audio). Cuenta con 14 pines de E/S digital (de los cuales 6 pueden utilizarse para salidas PWM y uno se puede utilizar para restablecer el módulo WT11), 6 entradas analógicas, un oscilador de cristal de 16 MHz, terminales para energía, una cabecera ICSP y un botón de reinicio. Contiene todo lo necesario para ser compatible con el microcontrolador y se puede programar de forma inalámbrica a través de la conexión Bluetooth. (Véase Figura 2).

Figura 2 Módulo Bluetooth para Arduino



Forma de comunicación por Bluetooth

El módulo Bluegiga WT11 en el Arduino BT proporciona una comunicación Bluetooth con computadoras, teléfonos y otros dispositivos Bluetooth. El ET11 se comunica con los ATmega328 vía serial (compartidos con los pines RX y TX de la placa). Viene configurado para

la comunicación 115200. El módulo es configurable y detectable por controladores Bluetooth de cualquier sistema operativo, el cual debe proporcionar un puerto COM virtual para su uso por otras aplicaciones. El software del Arduino incluye un monitor serial que permite a datos de texto simples ser enviados hacia y desde la placa Arduino través de esta conexión bluetooth. La placa también puede ser reprogramada utilizando esta misma conexión inalámbrica.

Comandos AT

Los comandos AT sirven para configurar el módulo Bluetooth a través de un microcontrolador, un ordenador o con cualquier dispositivo que posea una comunicación serie (Tx/Rx). Son instrucciones que permiten cambiar los baudios del módulo, el PIN, el nombre, etc. Para usar los comandos AT el módulo Bluetooth no debe estar vinculado a ningún dispositivo (led rojo del módulo parpadeando). Según las especificaciones del módulo, el tiempo que se tiene que respetar entre el envío de un comando AT y otro tiene que ser de 1 segundo. Si se envía un comando AT y en menos de un segundo se envía otro, el módulo no devuelve respuesta.

2.7 Relevador

El relevador es un interruptor que se puede activar mediante una señal eléctrica (enviada por el Arduino). Se podría decir que es un interruptor que conmuta entre dos contactos, pero también existen relevadores de múltiples contactos. Mediante una señal que controla al relevador se puede conmutar entre grandes tensiones e intensidades con sus contactos, los cuales cierran o abren los circuitos eléctricos, generando o interrumpiendo la conexión.

Sus principales características que conllevan a usar un relevador a otra tecnología para el uso de bombas son: Un relevador es capaz de soportar cualquier carga que se le ponga, solo basta con que los extremos metálicos de los contactos sean capaces de soportarlo.

Permite aislar el circuito de control del de la potencia, lo que es muy útil cuando se necesitan 2 diferentes tipos de tensión otorgando seguridad a diversas clases de dispositivos que requieren energía eléctrica.

Un relevador generalmente se utiliza cuando se requiere conmutar grandes picos de tensión o intensidad por ejemplo: para hacer funcionar motores de corriente alterna a cierta potencia (para este caso, una bomba de agua).

Debido a que se trabajará con Arduino, existe el módulo relay (relevador), para el hardware de Arduino que permite implementar esta tecnología para el prendido y apagado de bombas que se necesitara para el desarrollo del proyecto (Electroschematics, 2016).

2.8 Bomba

Existen muchos tipos de bombas, entre las cuales existen:

- Bombas de engranes.
- Bombas de aspa.
- Bombas de tornillo.
- Bombas de cavidad progresiva.
- Bombas de lóbulo.
- Bombas peristálticas.

Por mencionar algunas, pero la que se aplica en este proyecto es una bomba hidráulica sumergible; funciona de manera que se debe colocar dentro del líquido que se desea bombear. Su principal ventaja es que ofrece una fuerza de elevación significativa, pues no depende de la presión de aire para ascender el líquido (Mendez, L., 2011).

Posee sellos mecánicos que previenen que el líquido entre al motor causando un corto circuito; esta bomba es la ideal para este tipo de proyecto debido a que permite la posible reubicación del líquido anti plagas, sin afectar completamente la función del sistema ya instalado (mangueras, rociadores, etc.)

Es compatible con el módulo relevador del Arduino, lo que permite el acoplamiento de los componentes, permitiendo un uso óptimo del equipo.

3. DESARROLLO

Se diseñará, desarrollará e implantará un sistema móvil terrestre de fumigación para una invernaderos, ya que es estado de Yucatán es uno de los estados con mayor producción, pero suelen tener problemas de plagas; estos requiere de una solución a su problema de plagas, pues las pérdidas se han estado elevando debido a éstas que se van propagando día a día (Biodiversidad y desarrollo humano en Yucatán, (n.d)).

El sistema estará destinado para invernaderos de 6 a 12 metros de ancho por 30 a 40 metros de largo dedicados al cultivo chile habanero (*Capsicum Chinense* y sus variantes) la interfaz será por dispositivo móvil y el modo de transporte del sistema de riego será por un UGV o carrito fumigador.

3.1 Arquitectura base o candidata

La arquitectura del sistema estará integrada por los 6 elementos:

Interfaz: será desarrollada primordialmente para Smartphones con sistema operativo Android a partir de la versión 4.4 o superiores.

Módulo Bluetooth: Servirá de enlace y tráfico de datos hacia el microcontrolador.

Microcontrolador: El circuito integrado será capaz de ejecutar las órdenes ya previamente programadas. Este emitirá una señal que activará el módulo relay.

Módulo relay: Tiene la función de un interruptor, estará controlado por un circuito eléctrico en el que, por medio de una bobina y un electroimán, se acciona un juego de uno o varios contactos que permiten abrir o cerrar otros circuitos eléctricos independientes

Bombas y aspersores: la bomba servirá como propulsor, debido a que este succionara el insecticida y lo emitirá a través de las mangueras hasta llegar al aspersor.

Vehículo: El vehículo o UGV estará destinado para soportar y transportar el sistema así como la carga del contenedor de la sustancia a fumigar. Este contará con dos motores de 12v a 2.6ah y estará programado para seguir las instrucciones de la aplicación y del mapeo de la ruta.

Requerimientos Funcionales

- El sistema contará con un módulo de adquisición de información, donde se ingresará el tiempo a fumigar.
- El sistema contará con una interfaz de inicio
- El sistema contará con una sección para recomendaciones de fumigación y uso de insecticidas.
- El sistema permitirá al usuario ingresar la hora por medio del módulo de adquisición de información.
- El sistema contará con conexión Bluetooth
- El sistema permitirá al usuario conectar el móvil con el módulo Bluetooth.

No funcionales

- La programación de las fumigaciones se medirá por horas y minutos.

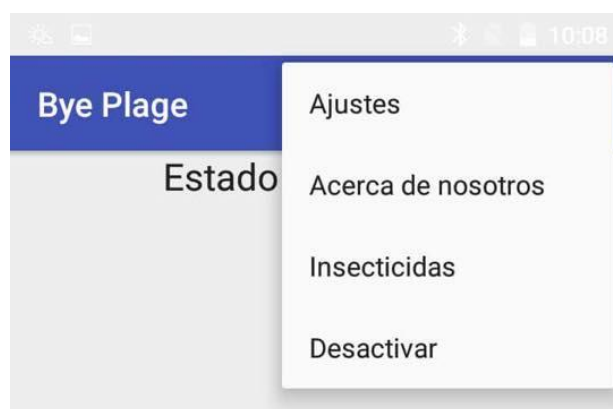
- La interacción será únicamente por vía remota, siempre y cuando se encuentre en el rango de cobertura del Bluetooth.
- La aplicación será solamente para versiones de Android 4.4 y/o superiores.

3.2 Interfaz de usuario

La interfaz fue diseñada en Android estudio ya que es una aplicación móvil dirigida primordialmente a usuarios de ese tipo de sistema operativo.

La aplicación tiene un diseño minimalista, en su pantalla de inicio se muestra un saludo de inicio así como el estado del fumigador. Una vez que se ingresa en la parte superior cuenta con un menú de opciones, el cual contiene ajustes: vista donde se busca establecer la conexión, acerca de nosotros, insecticidas: muestra una recomendaciones y tipos de insecticidas, desactivar. (Véase Figura 3)

Figura 3. Menú opciones



La sección "Acerca de" despliega una breve reseña de la empresa y su logo. (Ver Figura 4)

Figura 4 Acerca de



Una vez establecida la conexión con el dispositivo Bluetooth se visualiza un formulario básico en el cual se ingresa el horario a fumigar, además de los botones respectivos para sincronizar hora y agregar hora de fumigación. (Véase Figura 5)

Figura 5. Formulario



3.3 Construcción

La arquitectura que se utiliza se basa en la tecnología Bluetooth explicada anteriormente, que sirve de enlace entre el Smartphone y el microcontrolador. Una vez establecido el enlace se podrán programar las horas de fumigación a través de la interfaz gráfica creada en el IDE Android.

Con los datos y la conexión establecida, se envía una señal al microcontrolador y éste manda una señal al módulo relay que activa la bomba sumergible que está colocada dentro de un depósito cerrado donde se encuentra la sustancia para fumigar; la bomba pulsa esta sustancia por medio de mangueras al aspersor que la rocía sobre los plantíos del invernadero.

3.4 Pruebas de integración

Primera prueba de integración: Se integra el módulo Bluetooth con la aplicación móvil a fin de establecer la conexión entre dichos módulos. Se utilizar un Smartphone con la aplicación previamente instalada y se accede a ella, luego se verifica que el módulo Bluetooth del sistema de fumigación esté encendido (que el sistema esté encendido). En la aplicación se buscan los dispositivos cercanos y se accede al dispositivo Bluetooth (HC-05) para establecer la conexión, cuyo aviso de éxito es cuando un LED azul se enciende.

Segunda prueba de integración: Esta prueba realiza con éxito la conexión entre el arduino y el Bluetooth, que al completarse despliega el formulario para después agregar los horarios de fumigación. El tiempo es tomado en horas y minutos a modo de 24 horas.

Tercera prueba de integración: En esta prueba se realiza la conexión con el arduino, se agregan los horarios de fumigación y el encendido tanto de la bomba como de los motores.

Una vez que se tenga la conexión exitosa con el módulo Bluetooth y establecido el horario a fumigar, se espera a que llegue la hora establecida; los motores ya previamente conectados se encienden y el UGV empieza a dirigirse al lugar establecido.

4. Resultados

Primera iteración: constituye la prueba de los distintos módulos (aplicación, módulo Bluetooth, Arduino, módulo relays y bombas), esta prueba tuvo sus dificultades en cuanto al comportamiento deseado del sistema. En primera instancia no fumigaba en la hora establecida, ya que en el microcontrolador la hora predefinida no se visualizaba. En segunda instancia, al encenderse el sistema se ponía en funcionamiento los motores del carro y no se detenía.

Segunda iteración: en esta iteración se corrigieron los fallos de la primera, dando como resultado el buen funcionamiento tanto en el módulo de ingresar horario como en el encendido de las bombas y de los motores.

Tercera iteración: una vez corregidos los fallos detectados en las iteraciones pasadas se probó de nuevo el sistema teniendo un buen funcionamiento la aplicación desarrollada y en su conexión con el microcontralor, en el módulo que enciende las bombas y los motores.

Cabe señalar que la alimentación del carro fue cambiando en el transcurso de las iteraciones, en la primera y segunda etapa se tenía un acumulador de motocicleta pero se detectó que con esa alimentación solo se podían encender los motores dejando atrás la bomba. Una vez corregida la alimentación se probó con un acumulador de carro, permitiendo tener al sistema un tiempo de autonomía de poco más de cuatro horas.

5 Conclusiones

El sistema semiautónomo de carrito fumigador aquí descrito, en su versión temprana de prototipo, a pesar de que cumplía todos los requisitos funcionales, tuvo fallos inicialmente. En las iteraciones subsecuentes que llevan al prototipo final, se corrigieron todos los errores de funcionamiento, se redujo el tamaño, además de mejorar el tiempo de funcionamiento llegando a ser éste de 4 horas con 38 minutos, antes de la próxima carga de energía.

La aplicación móvil permite visualizar el contenido y su funcionalidad en la interfaz, una vez que se ha sincronizado con el sistema y así se tiene acceso a todas las funciones que éste permite.

La estructura del hardware y diseño del carrito fumigador permiten el uso del sistema en invernaderos locales, logrando así su objetivo, es decir, la fumigación semiautomatizada y la reducción del contacto humano con agroquímicos.

Es importante mencionar que para lograr la meta y el propósito de la primera etapa de este proyecto, se necesitó de un control estricto de los procedimientos y de 6 pruebas en promedio en cada módulo del hardware, cuyos resultados se han resumido en este trabajo.

6. Trabajo a futuro

Se planea realizar mejoras al prototipo en distintas partes incluyendo su arquitectura. Entre estas mejoras se pueden mencionar:

- Cambio de tamaño: esto se debe a que estaría dependiendo de cada invernadero, puesto que pueden cambiar los espacios de carriles y eso podría ser una desventaja para el UGV. Aunque por defecto los carriles de los invernaderos tengan una distancia que oscila entre 75cm y 1.5mt puede ser que sea menos la distancia.

- Mejoramiento de las llantas: el cambio de llantas para que sean de hule y tengan un mayor agarre. También si se encontrase en colinas, ya sea de subida o bajada, se podrían implementar orugas para el correcto movimiento y no perjudique el plazo de fumigación.
- Centro de carga: Aunque la capacidad de funcionamiento del carrito fumigador es de 4 horas con 38 minutos, tiempo suficiente para realizar hasta dos o tres fumigaciones dependiendo del tamaño de cada invernadero, se planea adaptarle un centro de carga y que cuando termine una jornada de fumigación, se desplace al centro de carga y empiece a acumular de nuevo carga hacia la batería.
- Múltiples boquillas: Las múltiples boquillas podrán utilizarse para fumigar más rápido y uniformemente, además como estas cuentan con distintos tipos de boquillas y cada una de ellas cuenta con un diferente tiro (chisguete). Esto dependerá del cliente y si requiere hacer el cambio y utilizar más boquillas.

BIBLIOGRAFÍA

Android Developers, (2012). *Android Developer..* Obtenido de Developer.Android:

<http://developer.android.com/intl/es/tools/studio/index.html>

Arduino. (2007). *Arduino CC*. Recuperado el 21 de Octubre de 2015, de Arduino CC:

<https://www.arduino.cc/en/Main/ArduinoBoardBT?from=Main.ArduinoBoardBluetooth>

Biodiversidad y desarrollo humano en Yucatán . (n.d). *Chiles cultivados en yucatan*. Chiles

cultivados en yucatan: Secretaría de Desarrollo Humano y Medio Ambiente

<http://www.seduma.yucatan.gob.mx/biodiversidad->

[yucatan/04Parte3_Usos_Biodiversidad/Capitulo7/04Chiles_cultivados.pdf](http://www.seduma.yucatan.gob.mx/biodiversidad-yucatan/04Parte3_Usos_Biodiversidad/Capitulo7/04Chiles_cultivados.pdf)

Bye Bugs. (2007), *Fumigacion profesional*. www.byebugsfumigaciones.com. Recuperado el 08 de mayo de 2016, de Bye Bugs.

Electroschematics. (2016). *Control a Relay with Arduino – Tutorial #5* Obtenido de

electroschematics: <http://www.electroschematics.com/8975/arduino-control-relay/>

Android Developers, (2012). *Android Developer..* Obtenido de Developer.Android:

<http://developer.android.com/intl/es/tools/studio/index.html>

Mendez, L. (2011). *Diferentes Tipos De Bombas, Aplicaciones*. Distrito Federal: UNAM.

Organización Mundial de la Salud. (2015). *OMS*. Recuperado el 19 de Octubre de 2015, de

OMS: <http://www.who.int/gho/publications/es/>

Trakür (2014), *Plataforma autoguiada para aplicaciones de fitosanitarios*. Recuperado el 15 de

05 de 2016, de presentacion de Trakür: [http://inta.gob.ar/sites/default/files/script-tmp-inta_-](http://inta.gob.ar/sites/default/files/script-tmp-inta_-_trakr_plataforma_autoguiada__hase_2013_gerardo.pdf)

[_trakr_plataforma_autoguiada__hase_2013_gerardo.pdf](http://inta.gob.ar/sites/default/files/script-tmp-inta_-_trakr_plataforma_autoguiada__hase_2013_gerardo.pdf)

Processing. (n.d). *Processing Org*. Recuperado el 20 de Octubre de 2015, de Processing Org:

<https://processing.org/>

Reas C. and Fry B. (2014), *Processing: A Programming Handbook for Visual Designers and Artists*, Second Edition, The MIT Press, 720 Pages, Hardcover.

Rip Mosquito & insects. (2006). *ripmosquito.com*. Recuperado el 08 de mayo de 2016, de
ripmosquito.com: <http://www.ripmosquito.com/qui%C3%A9nes-somos>

SAGARPA. (n.d). *El huerto familiar*. Recuperado el 08 de mayo de 2016, de El huerto familiar:
<http://www.sagarpa.gob.mx/desarrollorural/documents/fichasaapt/el%20huerto%20familiar.pdf>